

CLAIMS

1. A method operating a computer having a pipelined processor, comprising defining a bit within an instruction text field of a branch whereby to prevent the branch from being placed into a branch target buffer to thereby make the branch only detectable as the time frame of decode.
2. A method as defined in claim 1 comprising predicting the direction and target of a branch prior to decode.
3. A method as defined in claim 2 comprising predicting the direction and target of a branch prior to decode through a branch prediction array.
4. A method as defined in claim 1 comprising tracking the branch from the beginning of the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array.
5. A method as defined in claim 1 comprising denoting the instruction text field as a non-writable branch into the BTB.
6. A method as defined in claim 5 denoting the instruction field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked.
7. A method as defined in claim 5 denoting the instruction field in the non-system area, the branch may be predicted via aliasing.
8. A method as defined in claim 1 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer.
9. A method defined in claim 4 where branches which have targets that are highly non-constant can be blocked from branch predictions through the use the BTB blocking field in the instruction text.

10. The method as defined in claim 8 comprising denoting state altering code in the system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area.

11. A computer system having input, output, storage, and a pipelined processor, said processor adapted and configured to define a bit within an instruction text field of a branch whereby to prevent the branch from being placed into a branch target buffer to thereby make the branch only detectable as the time frame of decode.

12. A computer system as defined in claim 11, said computer system adapted and configured to predict the direction and target of a branch prior to decode.

13. A computer system as defined in claim 12 said computer system adapted and configured to predict the direction and target of a branch prior to decode through a branch prediction array.

14. A computer system as defined in claim 11, said computer system adapted and configured to track the branch from the beginning of the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array.

15. A computer system as defined in claim 11 said computer system adapted and configured to denote the instruction text field as a non-writable branch into the BTB.

16. A computer system as defined in claim 15 said computer system adapted and configured to denote the instruction field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked.

17. A computer system as defined in claim 15 said computer system adapted and configured to denote the instruction field in the non-system area, the branch may be predicted via aliasing.

18. A computer system as defined in claim 11 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer.

19. A computer system as defined in claim 14 where branches which have targets that are highly non-constant can be blocked from branch predictions through the use the BTB blocking field in the instruction text.

20. A computer system as defined in claim 18 said computer system is adapted and configured to denote state altering code in the system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area is prevented.

21. A program product comprising a storage medium having computer readable program code, said program code for use in a computer system having input, output, storage, and a pipelined processor, said program code adapting and configuring the computer system to define a bit within an instruction text field of a branch whereby to prevent the branch from being placed into a branch target buffer to thereby make the branch only detectable as the time frame of decode.

22. A program product as defined in claim 21, said computer system adapted and configured to predict the direction and target of a branch prior to decode.

23. A program product as defined in claim 22 said computer system adapted and configured to predict the direction and target of a branch prior to decode through a branch prediction array.

24. A program product as defined in claim 21, said computer system adapted and configured to track the branch from the beginning of the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array.

25. A program product as defined in claim 21 said computer system adapted and configured to denote the instruction text field as a non-writable branch into the BTB.

26. A program product as defined in claim 25 said computer system adapted and configured to denote the instruction field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked.

27. A program product as defined in claim 25 said computer system adapted and configured to denote the instruction field in the non-system area, the branch may be predicted via aliasing.

28. A program product as defined in claim 21 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer.

29. A program product as defined in claim 24 where branches which have targets that are highly non-constant can be blocked from branch predictions through the use the BTB blocking field in the instruction text.

30. A program product as defined in claim 28 said computer system is adapted and configured to denote state altering code in the system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area.

* * * * *